

Linux Networks Servers

PAM

PAM significa Pluggable Authentication Modules. Traduzindo: Módulos Anexáveis de Autenticação PAM é um conjunto de bibliotecas compartilhadas que permitem ao administrador do sistema local definir como as aplicações autenticam os usuários, sem a necessidade de modificar e recompilar programas.

Os softwares que utilizam o PAM para autenticação definem o seu próprio nome de serviço e o arquivo de configuração da política de autenticação. O software de login define um serviço chamado login e o seu arquivo de configuração da política de autenticação é /etc/pam.d/login.

O diretório /etc/pam.d é usado para armazenar arquivos de configuração da política de autenticação dos programas.

A sintaxe de cada arquivo encontrado nesse diretório segue o seguinte padrão:

```
<tipo_de_módulo> <palavra_de_controle> <módulo> <parâmetros>
```

Tipos de módulos: auth, account, password, session.

Para que serve o módulo auth?

Este módulo faz autenticação dos usuários, pedindo e verificando senhas e liberando o acesso ao sistema a estes.

Para que serve o módulo account?

Este módulo garante a autenticação, verificando se a conta do usuário em questão não expirou e se este pode acessar o sistema nos horários predeterminados.

Para que serve o módulo password?

Este módulo é usado para a criação de senhas e a sua configuração.

Para que serve o módulo session?

Este módulo é usado para gerenciar a sessão de um usuário que foi autenticado no sistema.

As palavras de controle são required, requisite, sufficient, optional.

Para que serve a palavra de controle required?

Este módulo deve ser verificado para permitir autenticação. Se ocorrer uma falha nesse módulo, o usuário não é notificado até que os outros módulos do mesmo tipo sejam verificados.

Linux Networks Servers

Para que serve a palavra de controle requisite?

Este módulo deve ser verificado para que a autenticação seja bem sucedida.

Entretanto, se esse módulo falha, o usuário é notificado imediatamente com uma mensagem indicando o que falhou.

Para que serve a palavra de controle sufficient?

A falha na verificação de um módulo não implica a falha da autenticação como um todo.
Mas se a verificação de um módulo marcado como sufficient for bem sucedida e nenhum módulo required tiver falhado, então os módulos restantes do mesmo tipo de módulo não são verificados e o usuário é autenticado.

Para que serve a palavra de controle optional?

A falha na verificação do módulo não implica a falha da autenticação como um todo.

A única ocasião em que um módulo marcado como optional é necessária para um autenticação bem sucedida é quando a verificação de nenhum outro módulo dessa classe falhou ou funcionou.

Neste caso, um módulo marcado como optional determina a autenticação para um módulo desse tipo.

Exemplos de módulos: pam_securetty e pam_nologin

Qual a função do módulo pam_securetty?

Este módulo verifica o terminal de login, não possui parâmetros e possui um arquivo de configuração: /etc/securetty.

Neste arquivo colocamos os terminais locais e remotos (por exemplo, telnet, pseudoterminais) a partir dos quais o usuário root pode fazer login. Ou seja, terminais seguros.

O módulo retorna sucesso para qualquer usuário que não seja o root e, se for root, somente se o terminal de onde está vindo o login estiver listado no arquivo /etc/securetty.

Um exemplo de pseudoterminal: Um terminal gráfico.

Abra uma instância do terminal gráfico!

Digite:

```
# echo "teste" > /dev/pts/0
teste
```

Linux Networks Servers

Qual a função do módulo pam_nologin?

Este módulo não possui parâmetros e desabilita o login de qualquer usuário que não seja o root.

Para isto, basta criar o arquivo /etc/nologin.
Se esse arquivo existir, o módulo pam_nologin retornará sempre erro para os usuários que não sejam o root.

Removendo o arquivo /etc/nologin, tudo volta ao normal, isto é, permite o login para usuários comuns.

Então, o que podemos fazer com o PAM?

O objetivo de se trabalhar com módulos do PAM, é possibilitar que as aplicações consigam fazer outros controles no login e na conta dos usuários, que por padrão não conseguiriam. Essa apostila tem um foco na questão da segurança, pois o PAM traz uma grande diversidade de módulos referente a esse assunto. Podemos utilizar um módulo do PAM que faz esse gerenciamento de horários de acessos dos usuários.

Podemos utilizar os módulos do PAM para várias outras coisas, como por exemplo: limitar usuário que podem logar, definir somente alguns grupos que podem fazer su no sistema, deixar as senhas mais seguras e muitas outras coisas. Os módulos do PAM podem ser utilizados para qualquer aplicação que tenha suporte ao PAM.

Como descobrir se uma aplicação tem suporte a PAM?

Basta utilizar o comando ldd.

Exemplo:

```
# ldd /bin/login
        linux-gate.so.1 =>  (0xb7f8b000)
        libpam.so.0 => /lib/libpam.so.0 (0xb7f53000)
        libpam_misc.so.0 => /lib/libpam_misc.so.0 (0xb7f4f000)
        libc.so.6 => /lib/tls/i686/cmov/libc.so.6 (0xb7df0000)
        libdl.so.2 => /lib/tls/i686/cmov/libdl.so.2 (0xb7dec000)
        /lib/ld-linux.so.2 (0xb7f71000)
```

Veja as linhas:

libpam.so.0 => /lib/libpam.so.0 (0xb7f53000)
libpam_misc.so.0 => /lib/libpam_misc.so.0 (0xb7f4f000)

Logo o login utiliza bibliotecas do PAM.

Linux Networks Servers

Parte prática:

Qual seria outra maneira?

Consultando o diretório /etc/pam.d:

```
# ls -l /etc/pam.d
```

Veja quais módulos do PAM estão disponíveis, consultando o diretório /lib/security:

```
# ls -l /lib/security
```

Como bloquear o login do root em terminais de modo texto sem o uso do PAM?

Como eu bloquearia o login do root em terminais de modo texto sem o uso do PAM?

Bloqueando o login do root em terminais de modo texto (sem o uso do PAM).

```
# vi /etc/securetty

tty1
# tty2
# tty3
# tty4
# tty5
# tty6
# tty7
# tty8
# tty9
# tty10
# tty11
# tty12
```

Basta apenas comentar a linhas do arquivo. Deixe pelo menos um terminal permitindo o root.

Agora, tente logar-se com o usuário root em um terminal, como, por exemplo, tty2, tty3 etc.

Como faríamos a mesma coisa só que agora com o uso do PAM?

Fazendo a mesma coisa, só que agora utilizando o PAM:

Descomente as linhas que estavam comentadas no arquivo /etc/securetty.

Linux Networks Servers

Agora, edite o arquivo /etc/pam.d/login e acrescente a seguinte linha:

```
account         requisite        pam_time.so
```

Edite o arquivo /etc/security/time.conf e acrescente a seguinte linha:

```
login;*;root;!Al0000-2400
```

Onde:

login - indica o serviço;
* - indica os terminais locais onde a política será aplicada;
root - a política será aplicada para o usuário root;
!Al0000-2400 - indica o horário permitido, sendo que o "!" é para não permiti-la naquele horário.

Sintaxe:

```
services;ttys;users;times
```

O primeiro campo indica o nome do serviço PAM que será controlado (um dos serviços contidos em /etc/pam.d).

O segundo campo indica lista de nomes de terminais que a regra que aplicará.

O sinal "&" tem a função and, "|" tem a função or e "!" especifica uma exceção.

O terceiro campo indica o nome de usuários afetados pela regra.

O quarto campo indica DiaSemana/faixa-de-horas que a restrição se aplicará.

O dia da semana é especificado em duas letras:

Mo - Segunda-feira
Tu - Terça-feira
We - Quarta-feira
Th - Quinta-feira
Fr - Sexta-feira
Sa - Sábado
Su - Domingo
Wk - Todos os dias da semana
Wd - Somente sábado e domingo (fim de semana)
Al - Todos os dias

Linux Networks Servers

O sinal "!" especifica uma exceção.

A faixa de horas é especificada após o dia no formato HHMM-HHMM.

O que significa MoTuWe0000-2400?

- Segundas, terças e quartas durante todo período de tempo (24 horas).

O que significa MoFrSu0800-1900?

Segundas, sextas e domingos das 08:00 da manhã às 19:00 da noite.

O que significa FrFr0500-0600?

Não será realizada na sexta (especificações repetidas são anuladas) de 05:00 as 06:00.

O que significa WkWe0838-1557?

Todos os dias da semana a partir de Quarta de 08:38 da manhã as 1557 da tarde.

O que significa AlMo0000-2400?

Todos os dias da semana, exceto segunda-feira.

Agora, vamos limitar os usuários que podem usar o su .

Crie uma política que não possibilite o uso do su, exceto pelos usuários do grupo admins:

```
# vi /etc/pam.d/su
 auth           required                  pam_whell.so group=admins
```

Dessa maneira, somente o usuário que estiver no grupo admins poderá ter acesso a fazer o su, mas por configurações do sistema.

```
# groupadd admins
# gpasswd -a leo admins
```

Edite o arquivo /etc/login.defs e descomente a seguinte linha:

```
SULOG_FILE   /var/log/sulog
```

Linux Networks Servers

Configure para ver o conteúdo do arquivo de log em tempo real no terminal 12:

```
# tail -f /var/log/sulog > /dev/tty12 &
```

Agora, utilize o comando su para alternar entre um usuário e outro:

```
# su toor
```

Veja o conteúdo do arquivo de log no terminal 12:

```
CTRL+ALT+F12
```

Como permitir acesso a grupos extras como, por exemplo, audio, floppy, cdrom etc?

Exemplo:

login;tty*;leo;Al0800-1900;floppy

Essa sintaxe quer dizer que o usuário leo só pode ter acesso ao grupo floppy se efetuar o login entre 08:00 da manha e 19:00 da noite.

Outro exemplo:

login;tty*;*;SaSu0000-2400;audio games

Essa sintaxe quer dizer que todos os usuários podem ter acesso ao grupo games e audio aos sábados e domingos.

Como limitar quais usuários podem logar?

Edite o arquivo /etc/pam.d/login e ative o módeulo pam_listfile.so da seguinte maneira:

```
# vi /etc/pam.d/login

auth    required       pam_listfile.so        item=user     sense=allow   file=/
etc/usuarios   onerr=fail
```



Agora crie o arquivo /etc/usuarios e coloque os nomes dos usuários que podem fazer login:

```
# vi /etc/usuarios
# Usuarios que podem fazer login
leo
```

Faça um teste e veja se o usuário leo pode logar e depois faça outro teste com um usuário comum.

Podemos utilizar algum módulo do PAM para limitar horários de acesso ao SSH?

Como posso fazer isso?

Pode-se utilizar o módulo pam_time.so para limitar horários de acesso ao SSH.

Primeiro edite o arquivo /etc/pam.d/ssh e adicione o módulo:

```
# vi /etc/pam.d/ssh
account          required          pam_time.so
```

Possibilite acesso via ssh somente no horário das 8:00 às 18:00:

```
# vi /etc/security/time.conf
ssh;*;!leo;Al0800-1800
```

Vamos falar agora sobre o pam_limits!

O arquivo de configuração do pam_limits é o /etc/security/limits.conf. Dentro dele, as linhas serão configuradas da seguinte forma:

<usuario/grupo> <tipo_de_limite> <recurso> <valor_do_limite>

Em usuário/grupo, podemos colocar um * para especificar todos os usuários, colocar um nome de usuário qualquer ou um nome de grupo, começando com @.

No tipo de limite, existem dois tipos possíveis: soft e hard. Quando o limite do tipo soft é chegado, o sistema avisa que chegou no limite mas não restringe nada. Se o limite for do tipo hard, o sistema simplesmente restringe o recurso e não deixa passar dalí. O soft então serve apenas para um “aviso” amigável, então na dúvida o hard é quem manda!

Linux Networks Servers

Exemplos de recursos:

core - Limite do tamanho do arquivo coredump (em KB). É a mesma configuração que obtemos com o comando “ulimit -c” da shell.

data - Tamanho máximo de segmento de dados que um programa pode usar. Também como: ulimit -d no bash.

fsize - Tamanho máximo de algum arquivo que o usuário possa criar. Também: ulimit -f.

memlock - Tamanho de memória alocada que os programas podem usar (em KB). Também: ulimit -l.

nofiles - Número máximo de arquivos abertos ao mesmo tempo. Também: ulimit -n.

rss - Tamanho máximo de memória compatilhada (em KB). Também: ulimit -m.

stack - Valor máximo de um processo executado (em KB). Também: ulimit -s.

cpu - Tempo máximo em minutos de uso da CPU. Também: ulimit -t.

nproc - Número máximo de processos executados ao mesmo tempo. Também: ulimit -u.

as - Limite em KB de espaço de endereçamento.

maxlogins - Número máximo de logins para esse usuário

maxsyslogins - Número máximo de logins no sistema

priority - Em qual prioridade padrão os processos desse usuário devem rodar.

locks - Número máximo de arquivos de locks que podem ser gerados pelo usuário. Também: ulimit -x.

nice - Número máximo de prioridade que o usuário pode setar, entre -20 (máxima prioridade) e 19 (mínima prioridade). Também: ulimit -e.

Como limitar que um usuário possa utilizar somente dois terminais consecutivos?

Vá até o diretório /etc/security, edite o arquivo limits.conf e insira a seguinte linha:

```
usuario          hard          maxlogins          2
```

Dessa forma, limita-se o usuário para utilizar somente dois terminais consecutivos.

Linux Networks Servers

Exemplo:

Limitando o tamanho máximo de um arquivo criado pelo usuário leo:

```
leo      hard     fsize    100
```

O usuário leo não pode criar arquivos maiores que 100KB (fsize 100).

```
$ dd if=/dev/zero of=arquivo bs=1024 count=100
100+0 records in
100+0 records out
102400 bytes (102 kB) copied, 0.0011873 s, 86.2 MB/s
```

Ele deixou criar um arquivo de 100KB. Agora vou criar um de 110!

```
$ dd if=/dev/zero of=arquivo bs=1024 count=110
File size limit exceeded
```